IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

|  |  |  |
|---|---|---|
| In re Application of: | § |  |
| James McKeeth | § | Group Art Unit: 2191 |
|  | § |  |
| Serial No.: 09/449,782 | § |  |
|  | § | Examiner: Steelman, Mary J. |
| Filed: November 26, 1999 | § |  |
|  | § |  |
| For: Command Line Output Redirection | § | Atty Docket: MICS:0194 |
|  | § | 99-3185.00 |

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313

CERTIFICATE OF TRANSMISSION OR MAILING
37 C.F.R. 1.8

I hereby certify that this correspondence is being transmitted by facsimile to the United States Patent and Trademark Office in accordance with 37 C.F.R. 1.6(d) or is being deposited with the U.S. Postal Service with sufficient postage as First Class Mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date below:

July 23, 2007
Date

Robert A. Manware

Dear Sir:

## PRE-APPEAL BRIEF REQUEST FOR REVIEW

In accordance with the Official Gazette Notice of July 12, 2005, Appellant respectfully submits this Pre-Appeal Brief Request for Review. This Request is being filed concurrently with a Notice of Appeal. In the Final Office Action Mailed March 23, 2007, the Examiner rejected claims 1-21 and 23-25. For at least the reasons set forth below, Appellant respectfully submits that the pending claims are allowable in their present form.

### Rejections under 35 U.S.C § 112

The Examiner rejected claims 1, 15, and 21 under 35 U.S.C. § 112, second paragraph, for "failing to clearly redefine the claim term and set forth the common definition so as to put one reasonably skilled in the art on notice that the applicant intended to so redefine that claim term." Final Office Action, p. 2. Appellant is unable to ascertain why the Examiner rejected the claims for "failure to clearly redefine the claim term...." Specifically, in explaining the rejection, the Examiner stated:

> The term "a call of a command line utility.., wherein the command line utility is a utility executable from a command line prompt" in claims 1, 15, and 21 is used by the claim to mean **"a utility that is capable of being executed from a command line prompt"** (prompt meaning some type of visual

> indicator, such as a flashing'>' character in a shell terminal interface or user interface), while the accepted meaning is literally **"a utility that is executed from a command line prompt."** The term is indefinite because the specification does not clearly redefine the term.
>
> Final Office Action, p. 5. (Emphasis added).

As set forth above, the Examiner asserts that both the definition used by the claims and the Examiner's "accepted meaning" are identical. Thus, Appellant requests clarification or withdrawal of the rejection of claims 1, 15, and 21 under 35 U.S.C. § 112.

If the Examiner maintains the rejection under 35 U.S.C. § 112, Appellant continues to maintain that the terms "command line utility" and "command line prompt" should be interpreted to have their ordinary and customary meaning as understood by a person having ordinary skill in the art. Appellant is not redefining the terms, and therefore requests that the Examiner define the terms "command line utility" and "command line prompt" according to their ordinary and customary meaning to a person having ordinary skill in the art.

**Rejections under 35 U.S.C § 103**

The Examiner rejected claims 1-21 and 23-25 under 35 U.S.C. § 103(a) as being unpatentable over Buxton (U.S. Patent No. 6,182,279, hereinafter "Buxton") in view of Qureshi (U.S. Patent No. 5,758,154, hereinafter "Qureshi"). Appellant respectfully traverses these rejections.

Embodiments of the present technique are directed to a method to provide command line utility output to an application without using temporary files. Specification, page 2. Specifically, each of the independent claims recite, *inter alia*, a "command line utility", an application providing "an identifier in the call" of the command line utility, and storing or having a "system storage" having a location to store the output of the command line utility.

**The cited references do not disclose any of the above claim features as recited in independent claims 1, 15, and 21.**

As preliminary matter, Appellant asserts that the Examiner's taking of Official Notice that "an executable command to modify the registry, is capable of being executed from a command line prompt" is inappropriate and unsupported. As stated in M.P.E.P. 2144.03, "[o]fficial notice without documentary evidence to support an examiner's conclusion is permissible only in some circumstances. While 'official notice' may be relied on, these circumstances should be rare when an application is under final rejection...Official notice unsupported by documentary evidence should only be taken by the examiner where the facts asserted to be well-known, or to be common knowledge in the art are capable of instant and unquestionable demonstration as being well-known." M.P.E.P. 2144.03. Thus, where there is a final rejection and the Examiner has provided no documentary evidence that a command to modify the registry is executable from the command prompt, Appellant believes that the Official Notice is unsupported and therefore inappropriate at this time. Appellant requests documentary evidence showing that "an executable command to modify the registry, is capable of being executed from a command line prompt," as argued by the Examiner.

Regarding the first § 103(a) reference, Appellant respectfully submits that Buxton does not teach, suggest, or disclose, either inherently or explicitly, "a command line utility." The Examiner admits that Buxton "failed to specifically disclose a 'command line utility,'" yet the Examiner states that Buxton "inherently invokes a utility." Final Office Action, pp. 7-8. In fact, Buxton does not disclose, either explicitly or inherently, a "utility," regardless of whether such "utility' is capable of being executed from the command line. The Examiner cited "WIN32 APIs" that work with "OLE libraries" as a "utility." In contrast, as *clearly and explicitly* stated in Buxton, APIs are "application program interfaces." Buxton, col. 7, lines 59-60. Further, as stated *clearly and explicitly* in Buxton, OLE libraries are "system-level services in accordance with the OLE specification 2.0." *Id.*, col. 8, lines 6-8. As would be clear to a person having ordinary skill in the art, "application program interfaces" and "system-level services" are quite different than a "utility," especially a "command line utility." Thus, Buxton does not disclose a "utility," or a "command line utility" as recited in independent claims 1, 15, and 21. Accordingly, Appellants assert that claims 1, 15, and 21 are allowable over Buxton, as the Examiner's arguments rely on Buxton disclosing a "utility" in combination with the unsupported Official Notice discussed above.

Additionally, Appellant disagrees with another characterization of Buxton made by the Examiner. The Examiner stated:

> Buxton suggested receiving commands via command line, which results in modifying the registry (system storage) and storage. Buxton failed to specifically disclose a "command line utility". Buxton suggests that the command line input (an

> object that consists of modifications to base component) is **directed (DIR utility — a command utility) to storage**, and the registry is edited (a command utility), but did not explicitly disclose 'command line utility'.
>
> Final Office Action, p. 8. (Emphasis added).

Appellant respectfully disagrees that Buxton suggests receiving "commands via a command line" that modify the registry. As stated above, Buxton discloses modifying the registry through the use of WIN32 APIs and OLE libraries, which are quite different than "commands via a command line." Further, the Examiner relied on the argument that the action in Buxton of "directed to storage" discloses a "DIR utility — a command utility." As argued in the Response to Final Office Action, the DIR utility *does not* "direct to storage" as the Examiner suggested, but instead is a directory command in Microsoft WINDOWS that lists the contents of a directory in the file system. Response to Final Office Action mailed May 23, 2007, p. 12. In the Advisory Action mailed June 18, 2007, the Examiner *admitted* that "Applicant is correct, DIR is a command to list directories." Thus, the Examiner admitted that the "directed to storage" action disclosed in Buxton is *not* the same as the DIR utility, and thus, is not a command line utility as argued by the Examiner. Accordingly, the Examiner's argument cited above that relies on this interpretation is not accurate, and Appellant requests the Examiner to withdraw the argument based on the admission in the Advisory Action.

Regarding the second § 103 reference, Qureshi, the Examiner stated:

> Qureshi clearly discloses (col. 3, lines 26-46) an application call to a registration routine ('command line utility') via system calls, to store an identifier in the registry. Col. 4, lines 2-7...With the registration DllRegisterServer routine, a new SRG tile that includes only the configuration information required by the new component can be distributed along with the new component, and the **registry can be properly updated by simply calling DllRegisterServer (a command line utility call) and supplying as an argument the pathname of the SRG file containing the configuration information required by the new component**." (emphasis added) Col. 5, lines 56-59,
>
> Final Office Action, p. 8. (Emphasis in original).

The Examiner asserted that both the "registration routine" and "DLLRegisterServer " disclose a "command line utility" and a "call of a command line utility" respectively. Appellant respectfully disagrees with this interpretation of Qureshi. In contrast to the Examiner's interpretation, Qureshi never mentions the terms "command line," "utility," or "command line utility." As *clearly stated* in Qureshi, the registration routine is implemented in a "registration DLL." Qureshi, col. 8, lines 1-3. As known to one having ordinary skill in the art, a DLL is a "dynamic link library." A DLL, and the routines accessible within a DLL, are far different than a "command line utility."

Although Appellant objects to the use of a Microsoft Computer Dictionary to define claim terms, as the claims encompass both Microsoft and non-Microsoft embodiments, Appellant will use the Microsoft Computer Dictionary in regards to terms in Qureshi as this dictionary was used by the Examiner. As defined in the dictionary used by the Examiner, a routine is "any section of code that can be invoked within a program." Microsoft Computer Dictionary 161 (5th 2005). Thus, the routines in a DLL are invoked by a program. As defined by the Examiner (using the same dictionary), a utility is "[a] program designed to perform a particular function; the term usually refers to software that solves narrowly focused problems or those related to computer system management." Final Office Action pp. 9-10. Thus, according to the dictionary preferred by the Examiner, if a utility is a program, and a routine is a section of code called *within* a program, then a routine is *clearly different* than a utility. Further, a routine is *clearly not* "a utility that is executed from a command line prompt," i.e. a "command line utility" as defined by the Examiner.

Accordingly, Buxton or Qureshi, taken alone or in combination, do not disclose each and every feature recited in the present independent claims. As such, whether taken alone or in combination, the cited references do not render obvious independent claims 1, 15, and 21 and the claims dependent therefrom.

Respectfully submitted,

Date: July 23, 2007

Robert A. Manware
Reg. No. 48,758
FLETCHER YODER
7915 FM 1960 West, Suite 330
Houston, TX 77070
(281) 970-4545